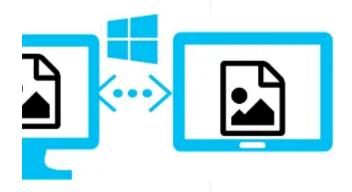
NAS'S BLOG



G A WIN32 HBITMAP TO WINRT/UWP SOFTWAREBITMAP

rent of machine learning and visual recognition it goes without saying that passing graphical information in more common scenario. This code snippet hopes to help the many queries that try to accomplish it in the sosystem. Especially going from Win32 to WinRT/UWP world.

```
: pd; // Capture picture info and plug the HBITMAP
RE picture; // Holds our picture data
1 stream; // Staging bridge buffer to WinRT/UWP
Buffer receiving bitmap data.
pRandomAccessStream;
pitmap buffer from COM API
. result = false;
:bSizeofstruct = sizeof(PICTDESC);
picType = PICTYPE_BITMAP;
>mp.hbitmap = hBitmap;
>mp.hpal = NULL;
SULT res = OleCreatePictureIndirect(&pd, IID_IPicture, false,
reinterpret_cast<void**&gt;(&amp;picture));
!SUCCEEDED(res))
return;
= CreateStreamOnHGlobal(0, true, &stream);
!SUCCEEDED(res))
picture->Release();
return;
bitmap buffer from COM API world to WinRT/UWP space.
iteRandomAccessStreamOverStream(stream, BSOS_DEFAULT, IID_PPV_ARGS(&spRandomAccessStream))
ve the RandomAccessStream populated it's vanilla WinRT walk in the park:
< BitmapDecoder^ &gt;
getDecoderTask(BitmapDecoder::CreateAsync((reinterpret_cast(spRandomAccessStream.Get()))));
ocontinuation = getDecoderTask.then([](BitmapDecoder ^bitmapDecoder) {
task getSoftwareBitmap(bitmapDecoder->GetSoftwareBitmapAsync());
getSoftwareBitmap.then([](SoftwareBitmap ^softwareBitmap)
     // softwareBitmap is your bitmap in WinRT/UWP World!
```

```
});
\ensuremath{\textit{I}}\xspace ait for the continuation to finish and handle any
error that occurs.
wcout <&lt; L"Waiting for tasks to finish..." &lt;&lt; endl;
continuation.wait();
\ensuremath{//} Alternatively, call get() to produce the same result.
//continuation.get();
:h (const exception& e)
wcout <&lt; L"Caught exception." &lt;&lt; endl;
                                                                  Powers Of The Dot Product >
ultiplayer Procedural Challenges & onvolving Feature Spaces
neme 404. Nas Amar
```